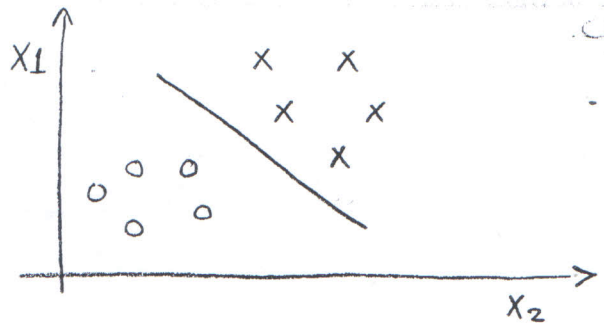
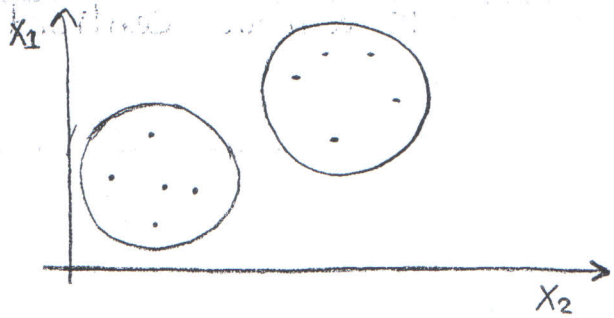
Lec : 8

## * UnSupervised Learning

For more information go to page 11

. given unLabeled training set

| Supervised Learning | Unsupervised Learning |
|---|---|
|   |   |
| Given Labeled training set $\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \dots, (x^{(m)}, y^{(m)})\}$ ($y^{(1)}$: right answer) | Given Unlabeled training set $\{x^{(1)}, x^{(2)}, x^{(3)}, \dots x^{(m)}\}$ there is no right answer. |

→ One of the most Famous Unsupervised learning algorithms is Clustering ALgorithm.

CLustering ALgorithm. group unLabeled data into different CLusters. (according to the relation between the data or determined thing)

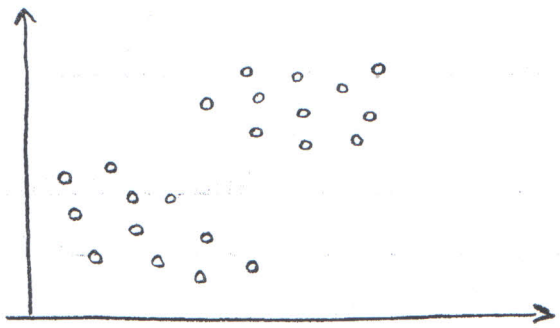
Applications OF Clustering
- Market segmentation
- Social network
- Organize Computing
- Astronomical data analysis

# Clustering

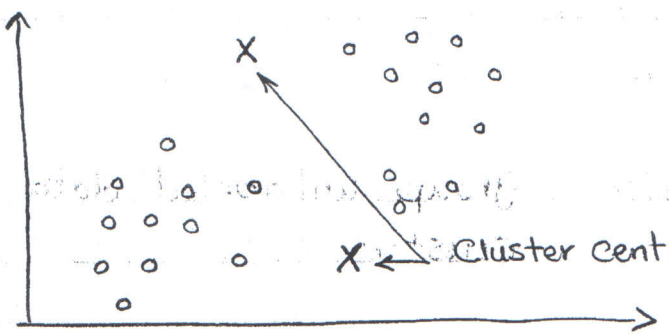
. K-means ALgorithm (mathematical algorithm)
group the unLabled data into different Clusters According to The distance between each data Location.

→ K Means is an iterative algorithm and it does two things. First is a cluster assignment step, and second is a move Centroid step.

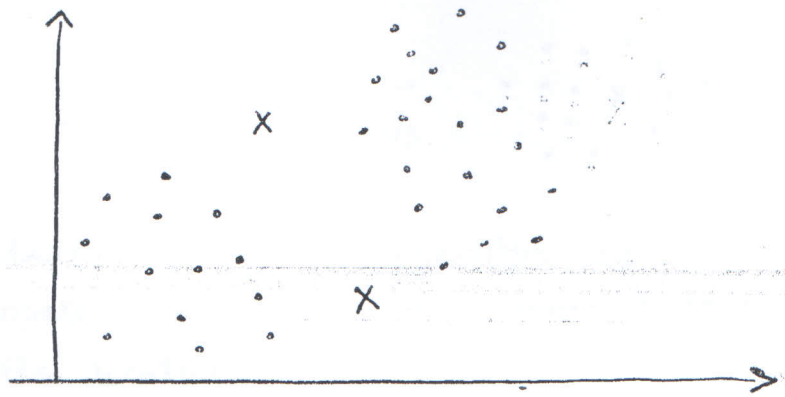
→ We have unLabled data



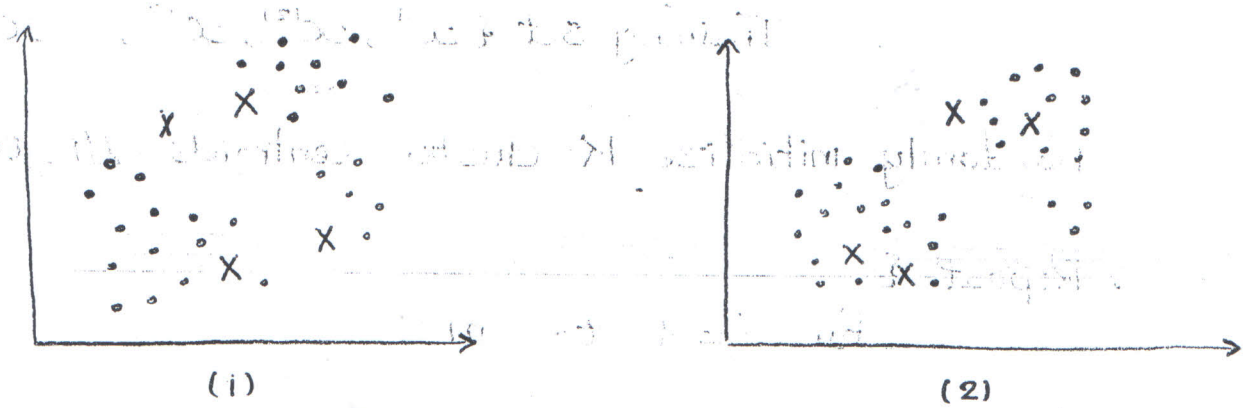
→ We select 2 cluster centroids in random.



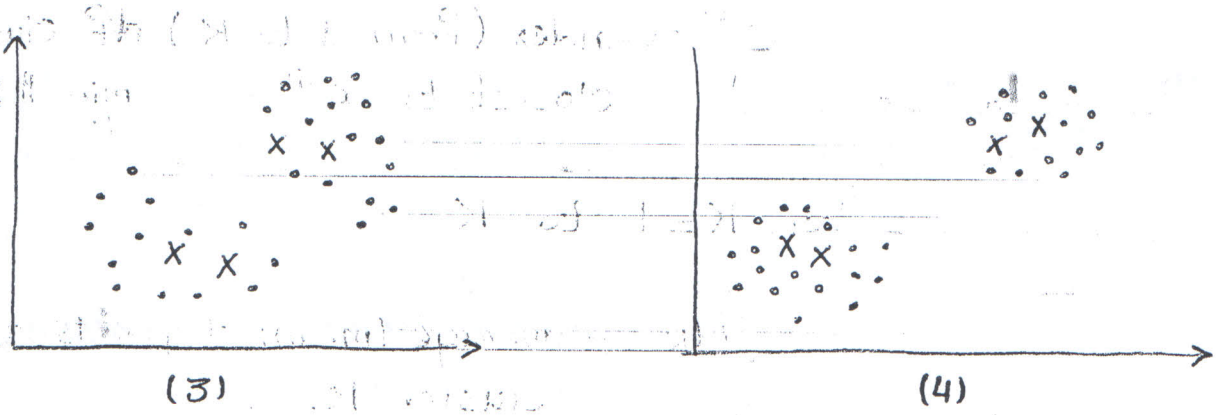
→ In the cluster assignment Step, the algorithm goes through each of the training examples, and depending on whether it's closer to which cluster centroid, it is going to assign each of the data points to one of the two cluster centroids.



→ In the move centroid Step, the algorithm is going to move the two cluster centroids to the average of the points of the same cluster.

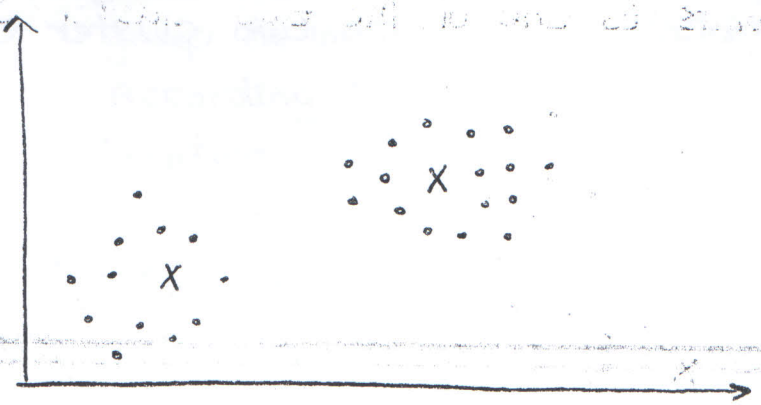


(1) (2)



(3) (4)

→ IF you keep running additional iterations of K means From here the cluster centroids will not change any further So at this point, K means has converged.



## K-means algorithm

. Input : - K (number of clusters (groups))
- Training set $\{x^{(1)}, x^{(2)}, x^{(3)}, \ldots x^{(m)}\}$

. Randomly initialize K cluster centroids $\mu_1, \mu_2 \ldots \mu_K \in$

. Repeat {

The Cluster assignment step → For $i = 1$ to $m$ (↓ number of training sets)

$c^{(i)}$ := index (from 1 to K (no. of clusters)) of cluster centroid closest to $x^{(i)}$ $\min_K \|x^{(i)} - \mu_K\|^2$

Number of cluster which the Point will belong to

→ choose cluster according to min distance between x and Cluster Centroi

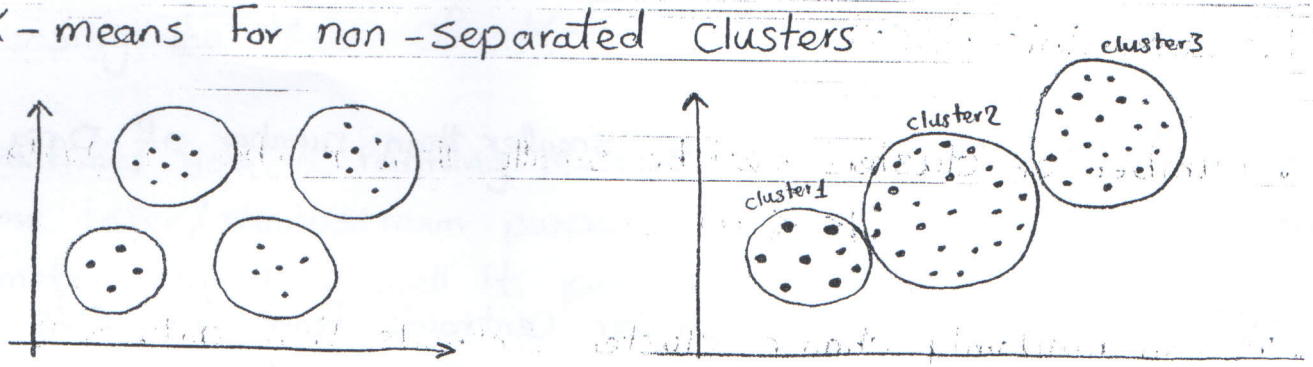
The move centroid step → For $K = 1$ to $K$

$\mu_K$ := average (mean) of points assigned to cluster K.

→ Choose new centroid location where the average of Sets assigned to this cluster.

}

## K-means For non-separated Clusters




Seperated Clusters  non-Separated Clusters

→ Using K-means algorithm for non-separated clusters we make sure that each set belonges to the nearest cluster from its location.

## K-means Optimization objective.

DiPs

$c^{(i)}$ : index of cluster $(1,2,\dots,K)$ to which example $x^{(i)}$ is currently assigned (cluster number)

$\mu_K$ : Cluster Centroid $K$ ($\mu_K \in \mathbb{R}^n$)

$\mu_{c^{(i)}}$ : Cluster centroid of cluster which example $x^{(i)}$ has been assigned (Centroid of cluster that $x^{(i)}$ will go to)

IF $c^{(i)} = 5 \Rightarrow \mu_{c^{(i)}} = \mu_5$

## Optimization objective

$$J(c^{(1)}, c^{(2)}, \dots, c^{(m)}, \mu_1, \dots, \mu_K) = \frac{1}{m} \sum_{i=1}^{m} \| x^{(i)} - \mu_{c^{(i)}} \|^2$$

Goal: $\min_{c^{(1)}, \dots c^{(m)}} J(c^{(1)}, \dots, c^{(m)}, \mu_1, \dots \mu_K)$

$\mu_1, \dots, \mu_K$ → This mean we must get each set belong to the cluster where the distance between $x^{(i)}$ and $\mu_K$ is the smallest

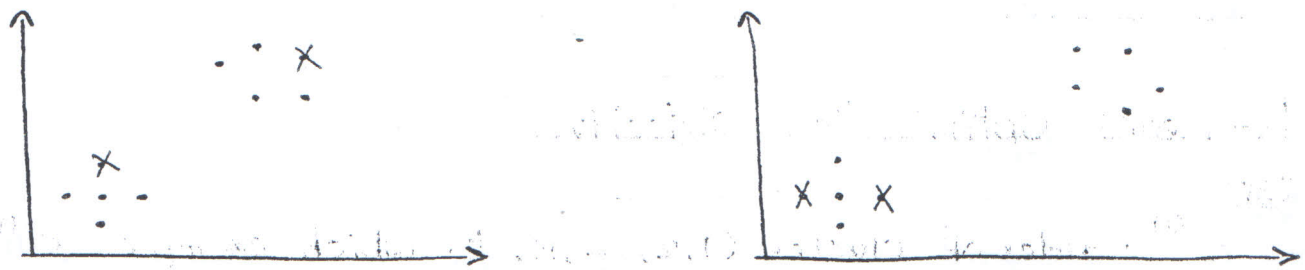
## Random initialization

- number of clusters must be smaller than number of Data sets

$$K < m$$

- We can randomly choose cluster centroids from data sets or Set them from our side of view.

- random initialization of cluster centroids is very important to get right answer or not



we get 2 clusters
(right Answer)

we can't get 2 clusters
(wrong Answer)

From this Example, You might really guess that K-means can end up converging to different solutions depending on exactly how the clusters were initialized.

For i = 1 to 100 {

Randomly initialize K-means.

Run K-means. Get $c^{(1)}, c^{(2)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K$

Compute Cost Function

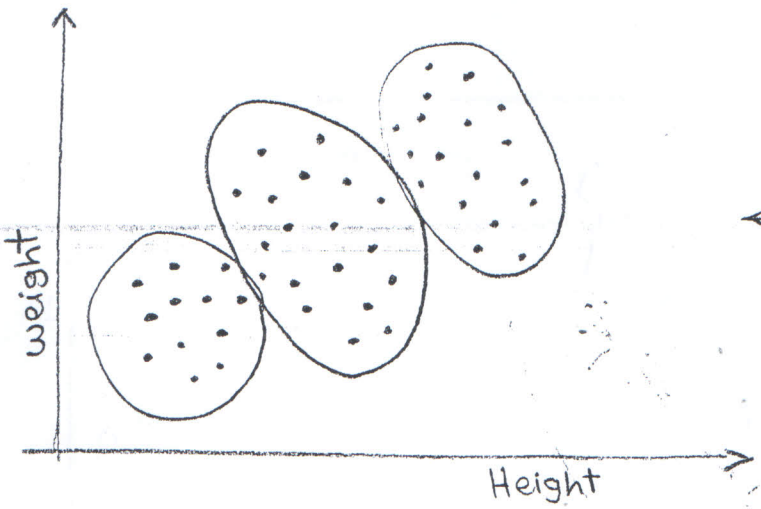
$J(c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K)$

}

Pick clustering that gave lowest Cost $J(c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K)$

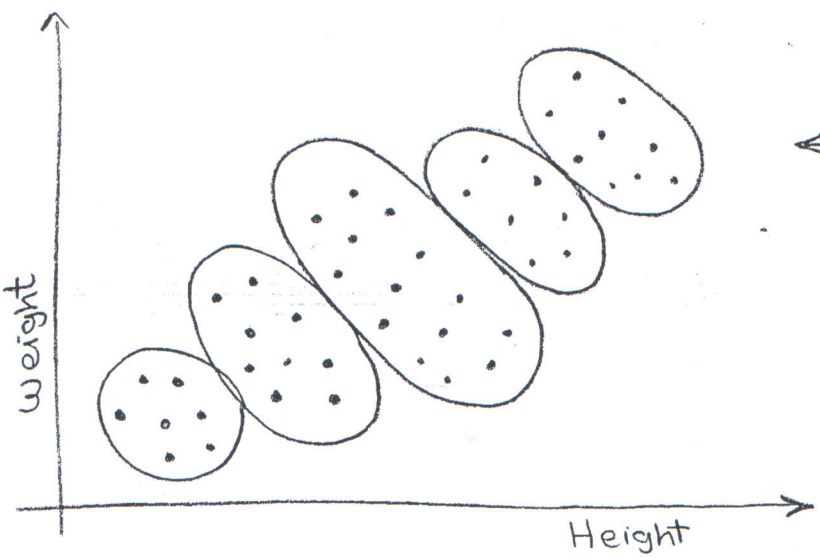
# Choosing The Value of K

[1] Sometimes, you're running K-means to get clusters to use for some later/downstream purpose. Evaluate K-means based on a metric for how well it performs for that later purpose.



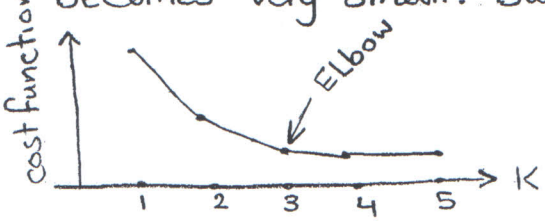

K: Cluster Number

← here, I want to group data into 3 clusters small, mid, Large




← here, I want to group data into 5 clusters, small, mid, large X Large, XX Large

[2] ELbow Method: (mathematical method)
determine number of clusters according to cost function, as we choose cluster number in which cost function change becomes very small. But this method causes wast in time.




we choose K = 3